LEVEL

Technical Report	574
	T. Bially R. Berger
Statistical Multiplexing of Speech in Hybrid Integrated Networks	
	16 June 1981

DTIC
ELECTE
SEP 1 1981

A

81 9 01 128

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

Raymond L. Loiselle, Lt.Col., USAF
Chief, ESD Lincoln Laboratory Project Office

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

LINCOLN LABORATORY

STATISTICAL MULTIPLEXING OF SPEECH
IN HYBRID INTEGRATED NETWORKS

T. BIALLY

Division 2

R. BERGER

Group 24

TECHNICAL REPORT 574

16 JUNE 1981

LEXINGTON   MASSACHUSETTS

ABSTRACT

This report describes a technique for achieving TASI- or DSI-like operation in hybrid integrated networks. End-to-end voice circuits are established separately for individual talkspurts along routes that are determined at dial-up and fixed for the duration of a conversation. The packet handling capability of the hybrid system is used for high-priority internode data communication regarding the onset and termination of speaker activity.

Subjects discussed include trade-offs between voice delay, cutout fraction and bandwidth efficiency that can be built into the nodal switching strategies. Analysis and simulation results are presented for cutout fraction versus voice delay in a talkspurt switching node with elastic buffering; the amount of buffer space needed as functions of TASI advantage and number of voice trunks per link; the probability of talkspurt buffering and average talkspurt delay. The report concludes with brief discussions of system vulnerabilities with respect to control errors and the compatibility of the concept with variable voice bit rate flow control schemes.

INTRODUCTION

Time-Assigned Speech Interpolation (TASI)[1] and Digital Speech Interpolation (DSI)[2] systems achieve approximate 2:1 bandwidth improvement over conventional telephony by exploiting the silence intervals in normal speech for the transmission of other voice signals. These methods are generally deployed over single links (e.g., submarine cables) in circuit switched networks that otherwise employ no statistical multiplexing. Since talkspurt and silence detection can only be performed on non-multiplexed speech streams, it is difficult to extend the concepts to multiple-link systems without introducing ancillary control information for use by downstream switches. Speech packetization formats afford a convenient means for conveying this information implicitly (i.e., packets are generated only during talkspurt) and, as a result, the concept of distributed statistical speech multiplexing has been promulgated mainly in the context of packet switching networks.[3] Statistical multiplexing gains in hybrid nets (circuit switched voice integrated with packet-switched data)[4] have centered on the use of speech silence intervals for the transmission of packetized data traffic (Time-Assigned Data Interpolation, or TADI) since this has a much less complicated control implication.

In this report we describe a technique for achieving TASI- or DSI-like operation in hybrid nets. In brief, the packet handling capability is used for high-priority internode communication regarding the onset and termination of talkspurts, which in turn are transmitted in a circuit-switched format. The result is that end-to-end voice circuits are

assigned separately to individual talkspurts but along routes that are fixed for the duration of a conversation. There is potential in this system for improved bandwidth efficiency relative to packet nets since the talkspurt/silence control overhead is amortized over entire talkspurts. On the other hand, voice cutout effects can be more severe than in a single link system and in this approach they will be concentrated at the beginning of a few talkspurts instead of being distributed over many speakers' entire utterances as they might be in a packet system. We propose to mitigate this effect by allowing network nodes to buffer incoming talkspurts until outgoing circuits become available. A related approach, as well as an excellent overview of delay related issues in voice transmission, is described in (5).

SYSTEM DESCRIPTION

Consider a multi-node net in which the wideband transmission links are organized into frames of N bits each, in a fixed TDMA fashion. A given voice connection C requires n_c of those bits in every link through which it is routed. The particular positions (not necessarily contiguous) of the bits are fixed in successive frame transmissions on a given link, but they may differ from link-to-link along the route as shown in Fig. 1. Small fixed delays (nominally one frame) are introduced at the switching nodes in order to accommodate these bit position assignments.

Voice routes through the net are determined at dial-up time in accordance with appropriate circuit routing procedures. However, time-slot assignments on the links along a given route are made separately for each talkspurt and the slots that are occupied are released when the
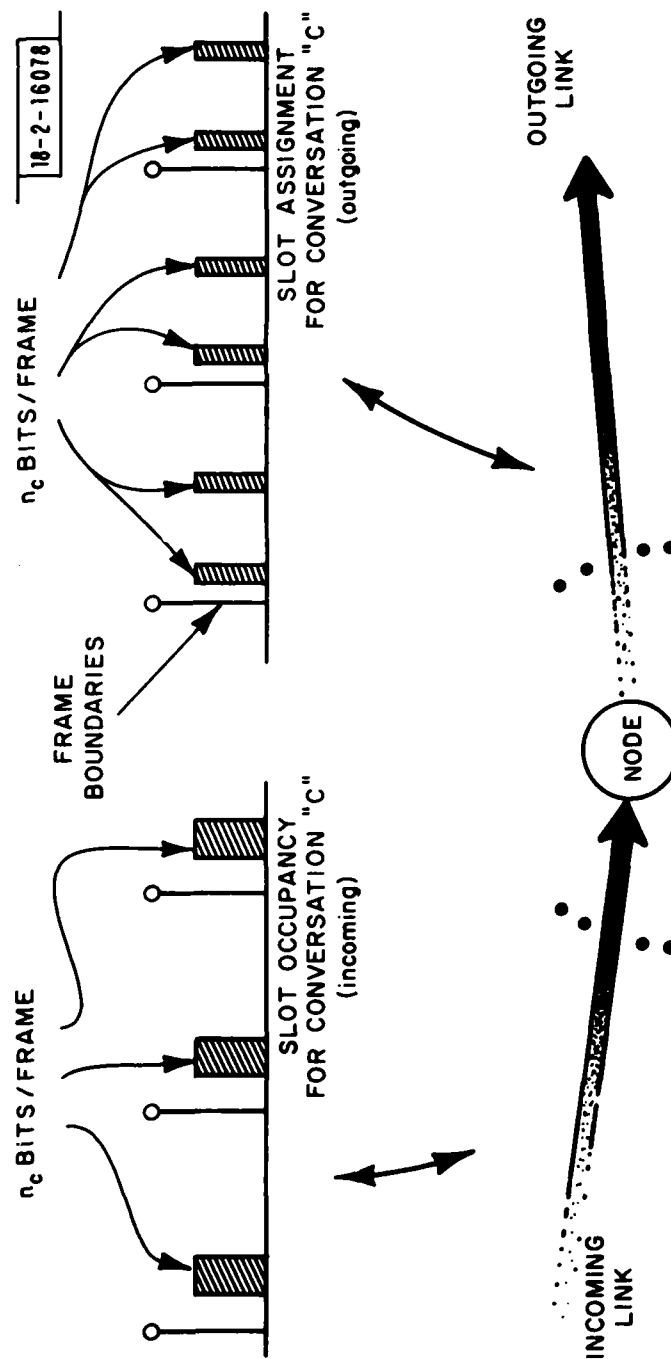
2

Fig. 1. Voice slot assignments in tandem links.

talkspurt terminates. They then become available for use by the talkspurts of other connections or for data packet transmissions. We assume for the time being that in each link all slots in a given frame that are not currently supporting talkspurt transmission can be aggregated and used for data packet transfers. The rate at which packet queues can "play out" into the network links will thus vary both with time and location in the net, as a function of instantaneous link voice load. Major issues in the design of a system of this type include:

    (a) A control structure for assigning link bandwidth to talkspurts and releasing it when the talkspurt is over.

    (b) Trade-offs between delay, TASI advantage, and cutout fraction that might be built into the nodal switching strategies.

    (c) The bandwidth efficiency that can be achieved, given that talkspurts are typically much longer than speech packets and might therefore amortize control overhead more effectively.

    (d) Mechanisms for aggregating non-voice slots and using them for packet data traffic on a dynamic, frame-by-frame basis.

These are elaborated upon below.

TALKSPURT CONTROL

Talkspurt/silence detection is performed at the node of origin or at the transmitting voice terminal. Talkspurts are modeled as constant bit-rate entities that require synchronous, clocked digital connectivity from source to destination. No connection is needed for silence. At the onset of a talkspurt, the originating node or terminal creates a

4

Talkspurt Control Packet (TCP) whose job it is to herald the appearance of the talkspurt to successive nodes along the selected route. Since timing will be critical for voice, TCPs may be treated preferentially with respect to other data traffic. A TCP might contain:

    (a) The ID of the virtual voice circuit to which it refers, and

    (b) The TDMA slots or frame bit positions within which the talkspurt will be arriving on the incoming link.

The job of a switching node upon receiving such a TCP is to refer to its routing tables and select the appropriate outgoing link for this talkspurt, identify a set of unused slots or bit positions in the outgoing TDMA frame into which the incoming bits will be transferred, modify item (b) of the TCP accordingly, and send the TCP packet ahead to the next node in the route. It also establishes, for the duration of the talkspurt, a circuit connection between the appropriate incoming and outgoing slots. A similar TCP is generated at the originating node at the completion of a talkspurt, for the purpose of releasing the slots that had been previously assigned. We thus add to the TCP contents a new item, i.e.,

    (c) Start or end of talkspurt.

PERFORMANCE TRADE-OFFS

The simple scenario described above will work as long as intermediate nodes can find the required number of free outgoing slots to accommodate newly arriving talkspurts. This will clearly not be the case at all times and some performance compromises will result. Three possibilities suggest themselves; i.e., a node can introduce cutout, add delay, or effect some combination of both. For the cutout case the node *simply*

ignores the incoming voice stream until it can forward it properly. The TCP is held until a suitable outgoing circuit is identified, and when that occurs, normal operation is resumed. The severity of the cutout phenomenon can be significant especially since it can happen more than once to the same talkspurt at different nodes along the route. One technique that has been suggested for dealing with this effect is to accord higher priority to talkspurts that have already been subjected to cutout by "upstream" nodes[6].

An alternative (or perhaps supplementary) method that we have studied in detail is for each node to buffer (instead of discard) its incoming talkspurts until outgoing circuits become available. Buffer delays introduced at successive nodes are additive and remain in effect for the entire talkspurt duration. Since excessive speech delays can be as undesirable as too much cutout, a balance between these two approaches might be in order. For example, one could add delay without introducing cutout until a predetermined maximum is reached, and then impose cutout without additional delay. Since both delay and cutout are additive as a talkspurt carves its way from one node to the next, it might be reasonable to include the following fields in the TCP:

   (d) Accumulated delay.

   (e) Accumulated cutout.

These fields would be modified at each switch and used by successive nodes as guides in assigning priority levels to the talkspurts or for deciding whether to apply delay or cutout if forwarding circuits are not immediately available.

6

Delay versus cutout characteristics for hybrid links of five and ten speech trunks operating with a TASI advantage (average number of talkers per trunk) of 1.6, are plotted in Fig. 2 along with similar results for pure packet switching systems operating under identical conditions. Results were obtained through computer simulation using average talkspurt and silence durations of 1.25 and 1.35 seconds, respectively, and following the speech activity model described in (7). In brief, the number of talkspurts in the system (being queued or transmitted) is described by a birth-death process in which the probability per unit time of a new talkspurt entering the system is proportional directly to the number of speakers currently in silence, and inversely to the average length of a silence. Similarly, the probability per unit time of a talkspurt leaving the system is proportional to the number of speakers currently in talkspurt and inversely to the average length of a talkspurt.

A somewhat less favorable cutout versus delay characteristic is evident for talkspurt switching relative to packet switching. This is due in part to the fact that when cutout and/or delay occurs in the talkspurt switched system, it is applied only to newly arriving talkspurts while those already in progress are allowed to continue without interruption. By way of contrast, cutout and delay effects are distributed evenly among the entire active speaker population in the packet switching approach. The probability that a newly arriving talkspurt will find all outgoing voice trunks busy, and therefore require buffering at a talkspurt switch, is plotted versus TASI advantage in Fig. 3.

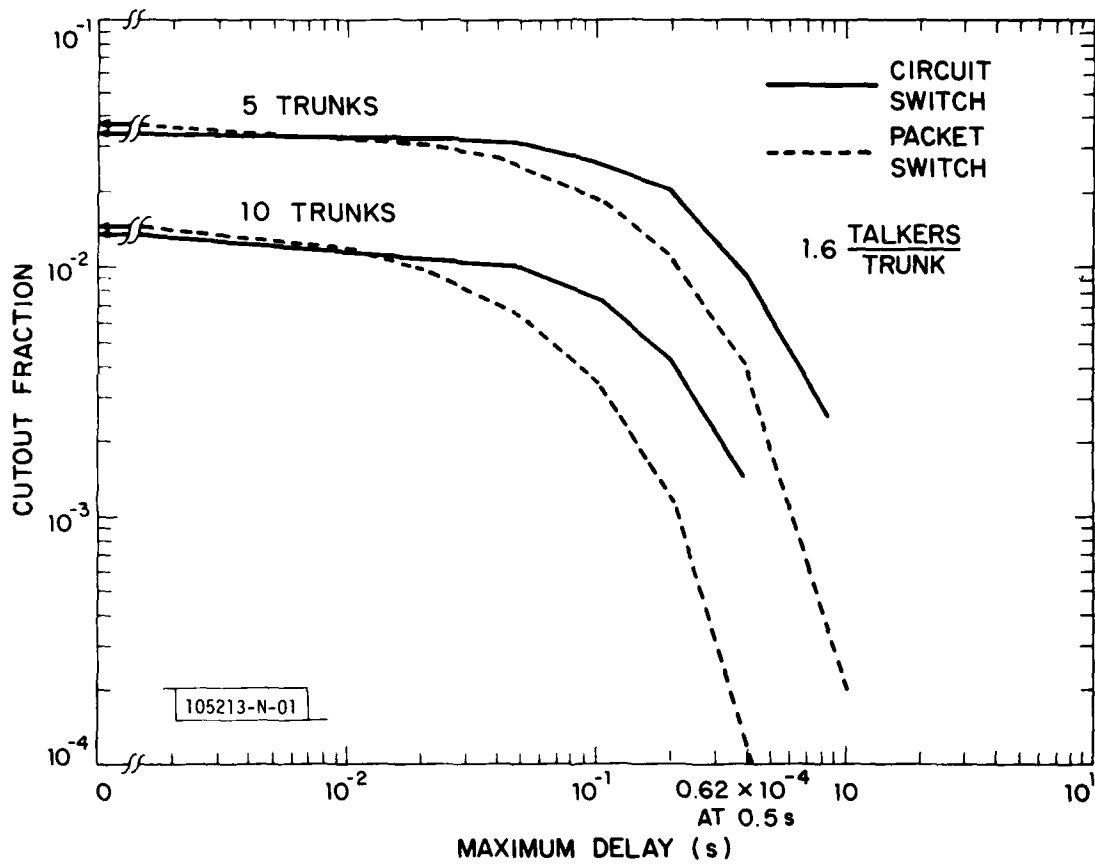Fig. 4 shows the average delay that will be experienced by a buffered

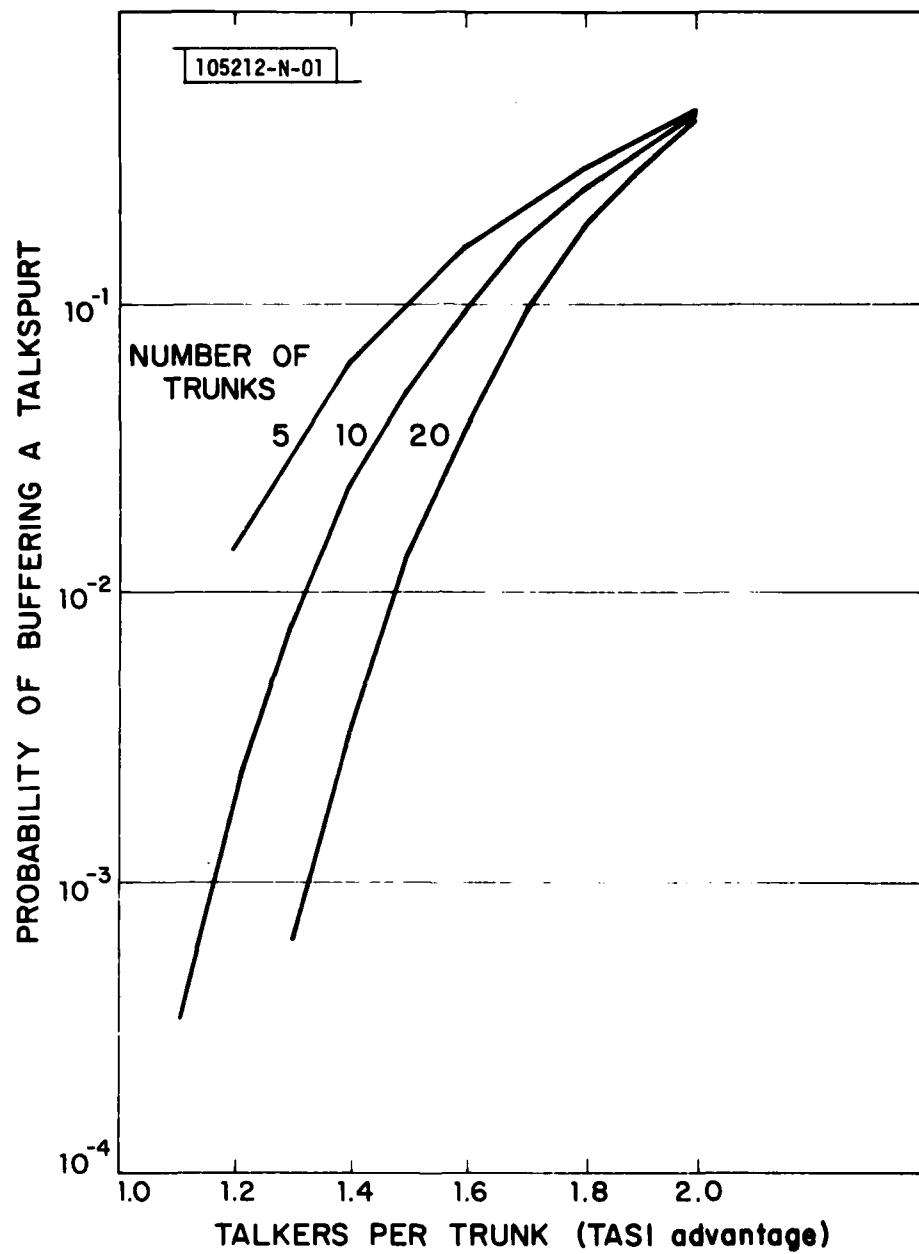Fig. 2. Voice cutout vs maximum buffer delay at a node.

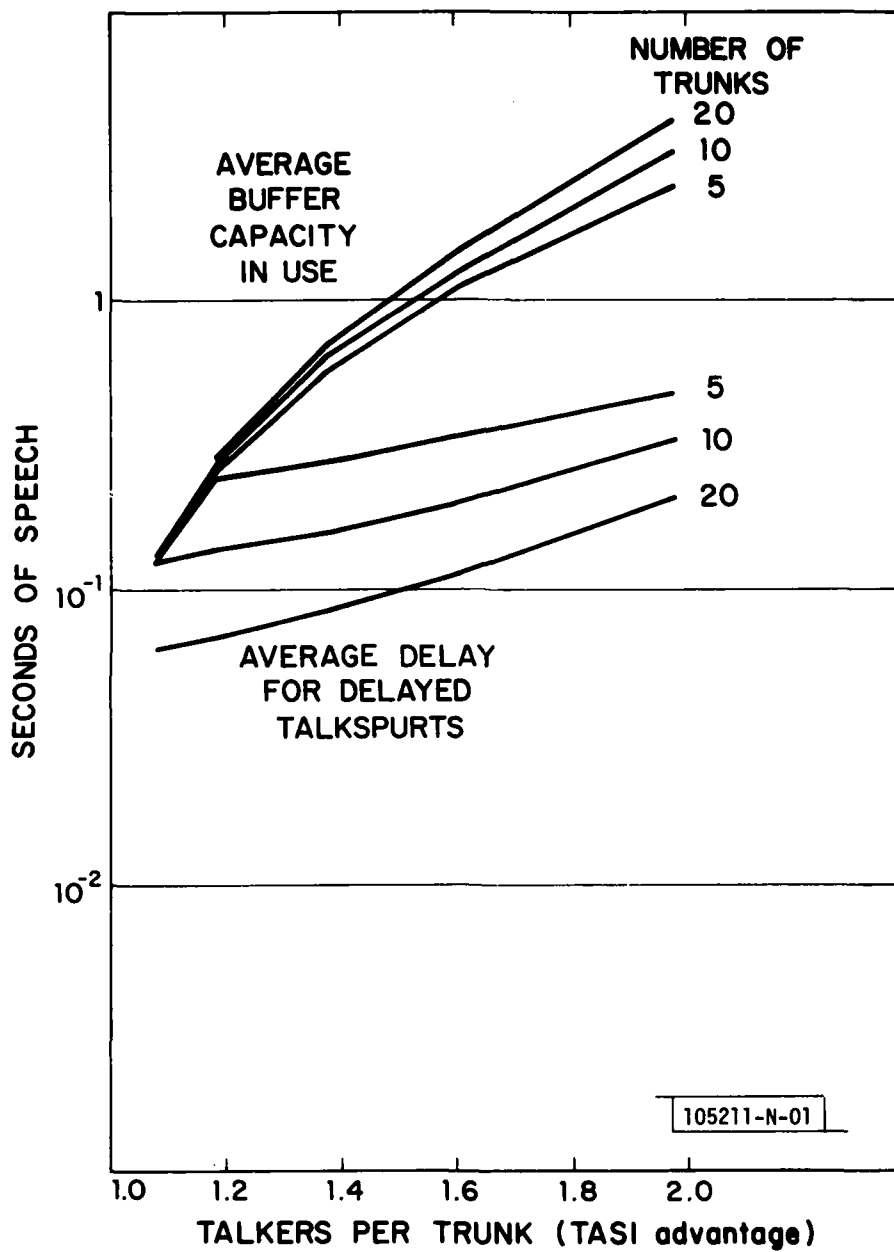Fig. 3. Probability of buffering vs TASI advantage.

9

Fig. 4. Buffer utilization and voice delay characteristics.

talkspurt and the average amount of buffer capacity in use at a talkspurt

switching node. It is interesting to note that the average required

buffer memory is several times the number of bits in the average (single

speaker) talkspurt, nearly independent of the number of outgoing voice

trunks. In addition, the average talkspurt delay, given that delay occurs,

is several times the time in which the combined output trunks can transmit

the number of bits in an average talkspurt. This is in the neighborhood

of 100 msec for the cases studied. The data of Figs. 2, 3 and 4 were

derived for a single link and some additional work is needed in order to

extend the results to tandem link configurations.

BANDWIDTH EFFICIENCY

An appealing aspect of this system is the potential that exists in

the average talkspurt for amortizing the bandwidth needed for TCP trans-

missions. If one thinks of a talkspurt as being a packet with a header

(TCP) and a trailer (another TCP), and compares this with more conventional

voice packets, the following emerges:

	64-kbps Voice		16-kbps Voice		2.4-kbps Voice	
	Packet	Talkspurt	Packet	Talkspurt	Packet	Talkspurt
Time Interval	20 msec	1.25 sec	20 msec	1.25 sec	40 msec	1.25 sec
Data Portion (bits)	1280	80,000	320	20,000	96	3,000
Header Portion (bits)	64	2 x 64	64	2 x 64	32	2 x 64
Efficiency (D/(D+H)) (%)	95.2	99.8	83.3	99.4	75.0	95.9

The above assumes that average talkspurts are of 1.25 second duration

and that packetization inefficiencies are reduced in the narrowband

(2.4 kbps) case through the use of abbreviated (32 bit) packet headers

and longer (40 msec) packet accumulation intervals. TCPs for all cases

11

are assumed to require the same number of bits as non-abbreviated voice packet headers, with two of these needed for each talkspurt. The bandwidth efficiency of talkspurt switching relative to packet voice is most pronounced at the lower voice bit rates where, despite short headers and longer accumulation intervals, the packetization process is relatively inefficient.

Although the TCPs themselves consume relatively little bandwidth in comparison to talkspurt transmissions, the obvious requirement that TCPs experience minimal queueing delays implies a signaling channel whose bandwidth exceeds the average TCP traffic load by a considerable amount. This could lead to reduced overall system efficiency in a voice-only network since a fair amount of data capacity would have to be set aside exclusively for TCP-handling purposes. Fig. 5 illustrates this point for the case of 48 voice users multiplexed onto a 24-circuit link (TASI advantage = 2). Note that although the average data rate of the TCP traffic is roughly 2.5 kbps, one requires a channel of 40 kbps capacity in order to reduce the average single-node queueing delays to less than one millisecond. Curves were derived by assuming that all TCPs are of equal length (64 bits) and that the lengths of talkspurts and silence intervals are exponentially distributed with equal mean durations of 1.25 seconds. These assumptions result in a standard M/D/1 queueing problem as described in (8).

As in the case of packet header amortization, the loss of network bandwidth efficiency due to signaling requirements becomes more pronounced at the lower voice bit rates. For instance, if 2.4kbps speech were used
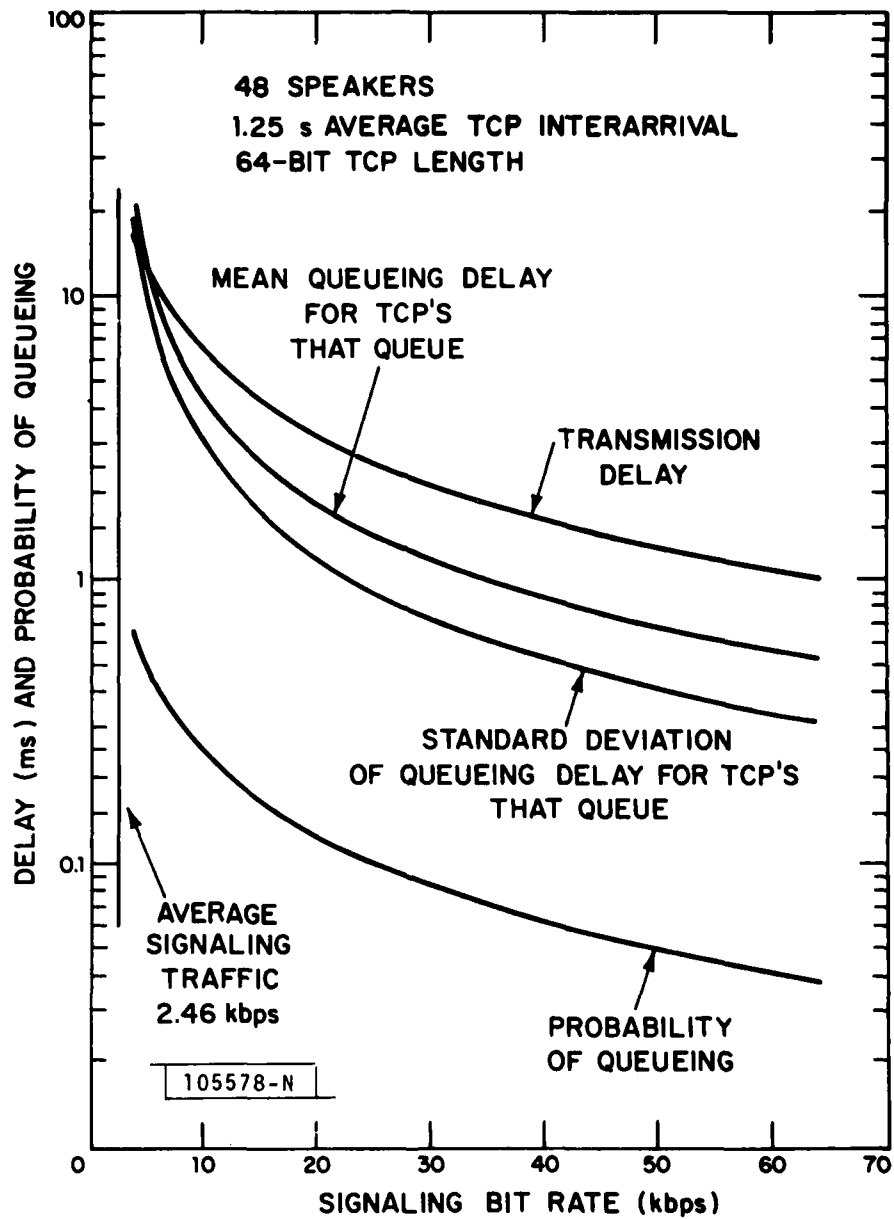
12

Fig. 5. Queueing and transmission delays for talkspurt control packets.

in the above example, the 40 kbps signaling channel would account for more than 40% of the total system bandwidth. However, the problem can disappear almost entirely in an integrated system in which a relatively large amount of data capacity is provided for general packet traffic. TCPs will basically "see" a virtual channel of bandwidth equal to the entire data allocation if they are allowed to bypass non-TCPs in packet queues. TCP queueing delays will thus be minimal but network bandwidth efficiency will remain high due to the utilization of the data channel by non-TCP packet traffic in the time intervals between TCP transmissions.

DATA TRANSMISSION

As in any integrated voice/data scheme, the objective here is to use all the bandwidth that is not committed to talkspurt transmissions for data packet transfers. Since packets will generally be of variable length, it would be nice to separate packet boundary and/or header issues from TDMA frame considerations. Referring to Fig. 6, we note that if we ignore frame boundaries and simply "blank out" those slots that are currently being used for voice, the result is the logical equivalent of a single synchronous clocked channel for data. This is basically the same notion that is used in the SENET[9] system.

The masking function (i.e., the blanking of voice bits) will in general be different for every link in the net, and will change with every TCP transmission. TCPs, in turn, are simply packets that flow on this dynamically changing data channel. Robustness is a major issue here and no simple solution is obvious. The basic problem is to assure that the two nodes at either end of a given link are always in agreement as to what
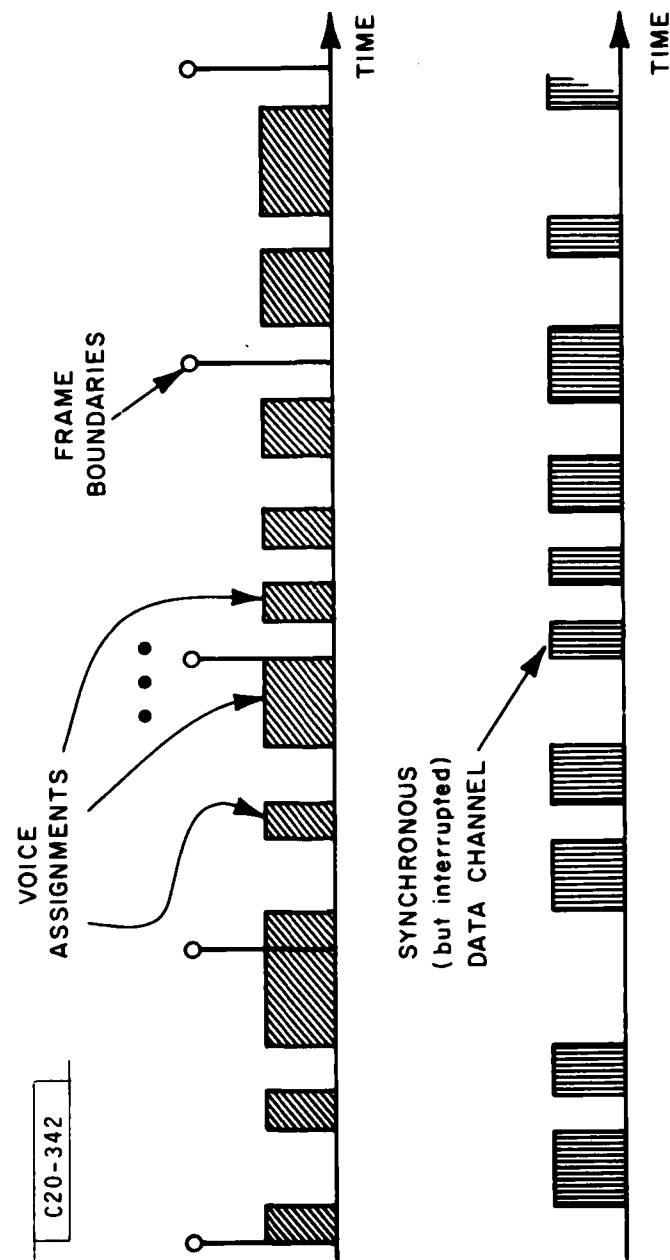
14

Fig. 6. Voice/data timing relationships.

C20-342

VOICE ASSIGNMENTS

FRAME BOUNDARIES

TIME

SYNCHRONOUS (but interrupted) DATA CHANNEL

TIME

15

bits belong to which user and for what purpose, and to provide simple and effective recovery mechanisms when they are not. A serious concern is whether a small set of transmission errors or nodal failures can wreak havoc with all the users on a link, or if the damage is restricted only to those users whose bits were directly affected.

Suppose a start-of-talkspurt TCP is received in error. That talkspurt will clearly be aborted and the listener will hear prolonged silence. However, when the next talkspurt for that conversation appears, assuming its TCP is not destroyed, the connection will be re-established. This example illustrates an interesting point; namely that in this system the talkspurts are somewhat like packets and the loss of one need not be felt by any others. Similarly, the disruption following the loss of a trailing TCP will last, at most, for the ensuing silence interval.

The robustness problem for data is more critical than for voice. With reference to Fig. 6, if a single voice TCP for any user on the link is received in error, a block of data bits corresponding to the associated voice circuit will be erroneously added or deleted from the data stream in every succeeding frame. Since TCPs are transmitted as data, there is the added danger that an error in one TCP can spawn errors in succeeding ones and cause catastrophic global problems. No simple answer is offered here. We simply observe that if all the voice slots are "compacted" to the beginning of each frame and if a count of the total voice allocation is sent with the frame, the data channel robustness problem is eased considerably. On the other hand, a single TCP error can now affect many voice users in many successive TDMA frames since slot allocations for

16

talkspurts are no longer fixed from frame-to-frame. A fairly clean answer might be to avoid compacting the speech but to quantize the slot sizes so that they all have M bits each. The leading bit in each slot could then flag whether it was carrying voice or data. Locations of the data bits are then unambiguously identified in each frame, and isolated mistakes cannot cause problems in future frames. The cost here is an a priori bandwidth efficiency limit of $(M - 1)/M$ plus whatever costs are associated with voice bit rates that are not exactly commensurate with an integer multiple of $(M - 1)$ bits per TDMA frame. Similar considerations will influence the choice of control strategy in almost any multi-user network design.

SUMMARY

The concept described here is very similar to the SENET notion in that a framed TDMA organization is used over wideband network links. The major difference is in the accommodation of TASI-like operation in the context of a multi-link network and the attendant increase in flexibility and bandwidth efficiency that this affords. We note that a multi-link TASI technique can very simply degenerate to a SENET-like system by disabling speech activity detection and viewing entire conversations as single talkspurts.

An important point worth emphasizing for this system is its potential compatibility with advanced voice-flow-control notions. For example, if embedded coding[10] were of interest, several parallel circuits could be established from source to destination, each carrying an embedded component of the voice stream. TCPs for these component circuits could

17

contain their relative priority numbers, and switching nodes could refuse to connect low-priority streams when overload conditions exist. Unlike a packet-oriented embedded coding scheme, this one would alter bit rates on a talkspurt-by-talkspurt basis, or impose greater leading-edge cutout on the less important bits. Potential perceptual problems due to instantaneous rate changes in mid-utterance would be avoided due to the built-in synchronization of these events with the talkspurt/silence boundaries.

REFERENCES

1. K. Bullington and J. Fraser, "Engineering Aspects of TASI," Bell Syst. Tech. J. 38, 353 (1959).

2. S. J. Campanella, "Digital Speech Interpolation," COMSAT Tech. Rev. 6, 127 (1976).

3. I. Gitman and H. Frank, "Economic Analysis of Integrated Voice and Data Networks: A Case Study," Proc. IEEE 66, 1549 (1978).

4. C. J. Weinstein, M. L. Malpass, and M. J. Fischer, "Data Traffic Performance of an Integrated Circuit-and Packet-Switched Multiplex Structure," Technical Note 1978-41, Lincoln Laboratory, M.I.T. (26 October 1978) DDC AD-A065449/1; also: IEEE Trans. Commun. COM-28, 873 (1980).

5. J. Gruber, "Delay Related Issues in Integrated Voice and Data Transmission: A Review and Some Experimental Work," 6th Data Commun. Symp., Pacific Grove, California, 27-29 November 1979, pp. 166-180.

6. M. Schwartz, K. Mase, and D. R. Smith, "Priority Channel Assignment in Tandem DSI," IEEE Trans. Commun. COM-28, 1802 (1980).

7. C. J. Weinstein, "Fractional Speech Loss and Talker Activity Model for TASI and for Packet-Switched Speech," IEEE Trans. Commun. COM-26, 1253 (1978).

8. L. Kleinrock, Queueing Systems, Vol. I: Theory (Wiley, New York, 1975).

9. G. Coviello and P. A. Vena, "Integration of Circuit/Packet Switching in a SENET (Slotted Envelope Network) Concept," presented at National Telecommun. Conf., Rec., 2, New Orleans, Louisiana (December 1975).

10. T. Bially, B. Gold, and S. Seneff, "A Technique for Adaptive Voice Flow Control in Integrated Packet Networks," IEEE Trans. Commun. COM-28, 325 (1980).

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS
BEFORE COMPLETING FORM

1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
ESD-TR-81-152	AD-A10 3 543	

4. TITLE *(and Subtitle)*	5. TYPE OF REPORT & PERIOD COVERED
Statistical Multiplexing of Speech in Hybrid Integrated Networks	Technical Report
	6. PERFORMING ORG. REPORT NUMBER
	Technical Report 574

7. AUTHOR(s)	8. CONTRACT OR GRANT NUMBER(s)
Theodore Bially and Robert Berger	F19628-80-C-0002

9. PERFORMING ORGANIZATION NAME AND ADDRESS	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
Lincoln Laboratory, M.I.T. P.O. Box 73 Lexington, MA 02173	Program Element No. 33126K

11. CONTROLLING OFFICE NAME AND ADDRESS	12. REPORT DATE
Defense Communications Agency 8th Street and So. Courthouse Road Arlington, VA 22204	16 June 1981
	13. NUMBER OF PAGES
	24

14. MONITORING AGENCY NAME & ADDRESS *(if different from Controlling Office)*	15. SECURITY CLASS. *(of this report)*
Electronic Systems Division Hanscom AFB Bedford, MA 01731	Unclassified
	15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

18. SUPPLEMENTARY NOTES

None

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

integrated networks
statistical multiplexing

packet speech
digital speech interpolation

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

This report describes a technique for achieving TASI- or DSI-like operation in hybrid integrated networks. End-to-end voice circuits are established separately for individual talkspurts along routes that are determined at dial-up and fixed for the duration of a conversation. The packet handling capability of the hybrid system is used for high-priority internode data communication regarding the onset and termination of speaker activity.

DD `FORM 1 JAN 73` 1473 EDITION OF 1 NOV 65 IS OBSOLETE

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*